

IN THE CLAIMS:

Please cancel claim 14 without prejudice.

- 1 1. (Currently Amended) A system comprising:
2 a plurality of network resources ~~adapted~~ configured to process received block-
3 based protocol data access requests; and
4 a plurality of virtual servers each ~~comprising~~ allocated a logical partitioning of the
5 network resources to establish an instance of a multi-protocol server, each virtual server
6 configured to service the block-based data access requests by converting the block-based
7 protocol requests to appropriate file system data requests, each virtual server further
8 configured to ~~allowed~~ shared share access to resources of the file system; and
9 ~~a context data structure provided to each virtual server; associated with a the~~
10 context data structure including information pertaining to a security domain of ~~the~~ that
11 virtual server for each supported access protocol, to enable controlled access to the
12 allocated and shared resources of ~~the file system~~ that virtual server.
- 1 2. (Original) The system of claim 1 wherein the network resources comprise
2 network interfaces assigned to one or more network address resources.
- 1 3. (Previously Presented) The system of claim 1 further comprising storage media
2 configured to store information as units of storage resources, the units of storage
3 resources allocated among each of the virtual servers.
- 1 4. (Original) The system of claim 3 wherein the units of storage resources comprise
2 volumes.
- 1 5. (Original) The system of claim 3 wherein the units of storage resources comprise
2 qtrees.

- 1 6. (Previously Presented) The system of claim 3 further comprising an operating
2 system having a file system resource adapted to perform a boundary check to verify that a
3 request is allowed to access certain units of the storage resources on the storage media,
4 each virtual server allowed shared access to the file system and further adapted to create
5 virtual disks within the units of storage resources and wherein each of the virtual disks
6 associated with one or more of the virtual servers.
- 1 7. (Previously Presented) The system of claim 6 wherein the operating system
2 further comprises a user interface having a command set adapted to operate on virtual
3 disks, and wherein the command set executes within a context of a virtual server.
- 1 8. (Original) The system of claim 7 wherein the user interfaces comprises a
2 command line interface (CLI) adapted to support the command set.
- 1 9. (Previously Presented) The system of claim 8 wherein the CLI comprises a lun
2 command adapted to perform operations to a virtual disk associated with the context of
3 the virtual server.
- 1 10. (Previously Presented) The system of claim 9 wherein the lun command creates a
2 logical unit number on a file system associated with the server, the logical unit number
3 being associated with the context of the virtual server.
- 1 11. (Original) The system of claim 8 wherein the CLI comprises an igroup command
2 that generates a set of file system primitive for binding an initiator group to one or more
3 initiator addresses and wherein the initiator group is associated with the context of the
4 virtual server.
- 1 12. (Original) The system of claim 1 wherein the block-based protocol comprises
2 iSCSI.

1 13. (Original) The system of claim 1 wherein the block-based protocol comprises
2 FCP.

1 14. (Cancelled).

1 15. (Original) The system of claim 1 wherein the multi-protocol server is further
2 adapted to process data access requests in response to one or more file-level protocols.

1 16. (Currently Amended) A method for implementing a virtual server, the method
2 comprising the steps of:
3 ~~adapting~~ configuring a plurality of network resources to process received block-
4 based protocol data access requests;
5 allocating logical partitions ~~partitioning of~~ the network resources to establish a
6 plurality of virtual servers, ~~each comprising as an~~ instances of a multi-protocol server
7 configured to service the block-based data access requests by converting the block-based
8 protocol requests to appropriate file system primitives; and
9 providing a context data structure to each virtual server, the context data structure
10 including information pertaining to a security domain of ~~the that~~ virtual server ~~for each~~
11 ~~supported access protocol~~, to enable controlled access to the allocated and shared
12 resources of ~~the file system~~ that virtual server.

1 17. (Previously Presented) The method of claim 16 further comprising the step of
2 configuring storage media to store information as units of storage resources, the units of
3 storage resources allocated among each of the virtual servers.

1 18. (Original) The method of claim 17 wherein the units of storage resources
2 comprise volumes.

1 19. (Original) The method of claim 17 wherein the units of storage resources
2 comprises qtrees.

1 20. (Currently Amended) A computer readable medium containing executable
2 program instructions ~~for implementing a virtual server, the executable program~~
3 ~~instructions comprising program instructions for executed by a process, comprising:~~
4 program instructions that configure adapting a plurality of network resources to
5 process received block-based protocol data access requests;
6 program instructions that allocate logical partitions partitioning of the network
7 resources to establish a plurality of virtual servers, ~~each comprising as an instances of a~~
8 multi-protocol server configured to service the block-based data access requests by
9 converting the block-based protocol requests to appropriate file system primitives; and
10 program instructions that provide providing a context data structure to each
11 virtual server, the context data structure including information pertaining to a security
12 domain of ~~the that~~ virtual server ~~for each supported access protocol,~~ to enable controlled
13 access to the allocated and shared resources of ~~the file system that~~ virtual server.

1 21-23. (Cancelled).

1 24. (Currently Amended) A method, comprising:
2 receiving a block-based data access request from a client;
3 forwarding the request to a virtual server;
4 performing security checks on the request using a context data structure provided
5 to each virtual server, the context data structure including information pertaining to a
6 security domain of ~~the that~~ virtual server ~~for each supported access protocol,~~ to enable
7 controlled access to ~~the~~ allocated and shared resources of ~~the file system that~~ virtual
8 server;
9 converting the received block-based data access request to a file system data
10 access request;
11 in the event that the request passes the security checks, servicing the file system
12 data access request to generate a response; and
13 forwarding the generated response to the client.

25. (Currently Amended) A system, comprising:
a network interface to receive a block-based data access request from a client;
the network interface to forward the request to a virtual server;
the operating system to perform security checks on the request using a context
data structure provided to each virtual server, the context data structure including
information pertaining to a security domain of ~~the~~that virtual server ~~for each supported~~
~~access protocol~~, to enable controlled access to ~~the~~allocated and shared resources of the
~~file system~~that virtual server;
in the event that the request passes the security checks, a process to convert the
received block-based data access request to a file system data access request;
the process to service the file system data access request to generate a response;
and
the process to forward the generated response to the client.

26. (Currently Amended) A computer readable media, ~~comprising:~~
~~— said computer readable media containing~~ program instructions for
~~execution~~executed by on a processor, ~~for the practice of a method, the method~~
comprising:
program instructions that receive ~~receiving~~ a block-based data access request from
a client;
program instructions that forward ~~forwarding~~ the request to a virtual server;
program instructions that perform ~~performing~~ security checks on the request using
a context data structure provided to each virtual server, the context data structure
including information pertaining to a security domain of ~~the~~that virtual server ~~for each~~
~~supported access protocol~~, to enable controlled access to the shared resources of the file
system;
program instructions that convert ~~converting~~ the received block-based data access
request to a file system data access request;

15 ~~program instructions that service in the event that the request passes the security~~
16 ~~checks, servicing the file system data access request to generate a response in the event~~
17 ~~that the request passes the security checks; and~~
18 forwarding the generated response to the client.